

# No Rastro dos Dados:

## Cowboys na Saga pelo Ouro Digital

Entenda os básicos da análise de dados



python™



pandas

Por: Lucas Ribeiro



# Explorando a Eficiência do Python:

## Listas, Tuplas e Dicionários

Em Python, a manipulação de dados é uma tarefa central e eficiente. Três estruturas de dados fundamentais para essa manipulação são as listas, tuplas e dicionários. Neste e-book, vamos explorar cada uma dessas estruturas e entender como elas podem ser utilizadas na prática.



# 01

## Listas

As listas são estruturas de dados fundamentais em Python, permitindo armazenar coleções ordenadas de itens. São versáteis e podem conter elementos de diferentes tipos.



# Listas



O que são listas?

Listas são conjuntos ordenados de itens em Python. São declaradas usando colchetes [] e os elementos são separados por vírgula.

```
# Criando uma lista com diversos  
# tipos de dados  
  
misturado = [1, 'dois', 3.0, True]
```

# 02

## Tuplas

Tuplas são semelhantes às listas, mas com uma diferença crucial: elas são imutáveis, ou seja, seus elementos não podem ser alterados após a criação.



# Tuplas

O que são tuplas?

Tuplas são sequências ordenadas de elementos, semelhantes às listas, mas definidas por parênteses (). São úteis para dados que não devem ser alterados.

```
# Criando uma tupla de valores constantes  
cores = ('vermelho', 'verde', 'azul')
```

# 03

## Dicionários

Dicionários são estruturas de dados que permitem armazenar pares chave-valor. São eficientes para realizar buscas e associações de dados.



# Dicionários

O que são dicionários?

Dicionários são estruturas de dados que mapeiam chaves para valores. São definidos por chaves {} e contêm pares chave-valor separados por vírgulas.

```
# Criando um dicionário de contatos  
contatos = {'Alice': 12355, 'Bob': 55123}
```



# 04

## Pandas

O Pandas é uma biblioteca de software em Python usada para manipulação e análise de dados. Ela oferece estruturas de dados e ferramentas de análise de alto desempenho, tornando-a uma escolha popular para cientistas de dados, analistas e desenvolvedores.



# Pandas



O que é o Pandas?

O Pandas é uma biblioteca open-source que fornece estruturas de dados flexíveis e ferramentas para trabalhar com dados estruturados. Seu principal objetivo é facilitar a manipulação, limpeza e análise de dados em Python.

Por que usar o Pandas?

- Facilita a leitura e escrita de diferentes formatos de dados, como CSV, Excel, SQL, JSON, entre outros.
- Oferece ferramentas poderosas para limpar e preparar dados para análise.
- Permite realizar análises estatísticas e exploratórias de forma eficiente.
- É altamente flexível e extensível, com uma comunidade ativa que contribui com novos recursos e melhorias.

# 05

## DataFrames

DataFrames são estruturas de dados bidimensionais extremamente poderosas fornecidas pela biblioteca Pandas em Python. São semelhantes a tabelas de banco de dados ou planilhas Excel, permitindo manipulação e análise de dados de forma eficiente.



# DataFrames



O que são DataFrames?

DataFrames são estruturas de dados tabulares com linhas e colunas rotuladas. São construídos sobre a estrutura de Series do Pandas e são amplamente utilizados para análise e manipulação de dados.

```
import pandas as pd

# Criando um DataFrame a partir de um dicionário
dados = {'Nome': ['Alice', 'Bob', 'Charlie'],
         'Idade': [25, 30, 35],
         'Cidade': ['Rio de Janeiro', 'São Paulo', 'Belo Horizonte']}

df = pd.DataFrame(dados)

# Exibindo o DataFrame
print(df)
```



# 06

## Séries

---

As Séries são estruturas de dados unidimensionais fornecidas pela biblioteca Pandas em Python. Elas são semelhantes a arrays unidimensionais, mas com capacidades adicionais e funcionalidades otimizadas para análise de dados.



# Séries

O que são Séries?

Séries são estruturas de dados unidimensionais rotuladas em Pandas. Elas são construídas sobre a estrutura de arrays NumPy, mas com recursos adicionais, como rótulos de índice e manipulação integrada de dados ausentes.

```
import pandas as pd

# Criando uma Série a partir de uma lista
valores = [10, 20, 30, 40, 50]
serie = pd.Series(valores)

# Exibindo a Série
print(serie)
```



**ESSE É O FIM,  
PARCEIRO!**

Agradecimentos logo em seguida!



# Obrigado por acompanhar esta jornada!



O E-book que acabou de ler foi todo gerado por uma IA.  
O passo a passo se encontra no meu GitHub.

Por se tratar de uma obra com fins didáticos, nenhuma validação minuciosa foi realizada, portanto, erros podem acontecer.



[Repositório GitHub](#)



[Perfil da DIO](#)